Docket No. AUS920030469US1 ***PATENT***

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Gallagher et al.**	§	
	§	Group Art Unit: **2114**
Serial No. **10/616,848**	§	
	§	Examiner: **Truong, Loan**
Filed: **July 10, 2003**	§	
	§	
For: **Method and Apparatus for Managing Adapters in a Data Processing System**	§	
	§	

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

35525
PATENT TRADEMARK OFFICE
CUSTOMER NUMBER

APPEAL BRIEF (37 C.F.R. 41.37)

This brief is in furtherance of the Notice of Appeal, filed in this case on December 13, 2006.

A fee of $500.00 is required for filing an Appeal Brief. Please charge this fee to IBM Corporation Deposit Account No. 09-0447. No additional fees are believed to be necessary. If, however, any additional fees are required, I authorize the Commissioner to charge these fees which may be required to IBM Corporation Deposit Account No. 09-0447. No extension of time is believed to be necessary. If, however, an extension of time is required, the extension is requested, and I authorize the Commissioner to charge any fees for this extension to IBM Corporation Deposit Account No. 09-0447.

REAL PARTY IN INTEREST

The real party in interest in this appeal is the following party: International Business Machines Corporation of Armonk, New York.

RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in the pending appeal, there are no such appeals or interferences.

STATUS OF CLAIMS

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1-22.

B. STATUS OF ALL THE CLAIMS IN APPLICATION

1. Claims canceled: None.
2. Claims withdrawn from consideration but not canceled: None.
3. Claims pending: 1-22.
4. Claims allowed: None.
5. Claims rejected: 1-22.
6. Claims objected to: None.

C. CLAIMS ON APPEAL

The claims on appeal are: 1-22.

STATUS OF AMENDMENTS

No amendments were submitted after the Final Office Action of September 13, 2006.

SUMMARY OF CLAIMED SUBJECT MATTER

A. CLAIM 1 - INDEPENDENT

The subject matter of claim 1 is directed to a method of in a device driver for handling a failure of a primary adapter in a data processing system. (Specification, p. 3, ll. 3-5; p. 11, l. 26-p. 13, l. 9; figs 3 & 4, all elements). The method includes queuing data in a data queue used by the primary adapter. (Specification, p. 9 ll. 20-25; p.11, ll. 19-20; fig. 3, step 316). The method further includes monitoring the primary adapter for the failure. (Specification, p. 10, l. 24; p. 12, ll. 10-15; figure 4, step 404). The method further includes, responsive to detecting the failure, switching to a standby adapter handled by the device driver, wherein the standby adapter uses the data in the data queue. (Specification, p.9, l. 5-p. 10, l. 22; p. 12, l.18-p. 13, l. 9).

B. CLAIM 8 - INDEPENDENT

The subject matter of claim 8 is directed to a data processing system for handling a failure of a primary adapter in a data processing system. (Specification, p. 3, ll. 3-5; p. 11, l. 26-p. 13, l. 9; figs 3 & 4, all elements). The data processing system includes a queuing means for queuing data in a data queue used to the primary adapter (Specification, p. 9 ll. 20-25; p.11, ll. 19-20; fig. 3, step 316). The data processing system further includes a monitoring means for monitoring the primary adapter for the failure. (Specification, p. 10, l. 24; p. 12, ll. 10-15; figure 4, step 404). The data processing system further includes a switching means for switching to a standby adapter handled by the device driver responsive to detecting the failure, wherein the standby adapter uses the data in the data queue. (Specification, p. 10, l. 24; p. 12, ll. 10-15; figure 4, step 404).

C. CLAIM 15 - INDEPENDENT

The subject matter of claim 15 is directed to a computer program product in a recordable-type medium for handling a failure of a primary adapter in a data processing system. (Specification, p. 3, ll. 3-5; p. 11, l. 26-p. 13, l. 9; figs 3 & 4, all elements). The computer program product includes first instructions for queuing data in a data queue used by the primary adapter. (Specification, p. 9 ll. 20-25; p.11, ll. 19-20; fig. 3, step 316). The computer program product further includes second instructions for monitoring the primary adapter for the failure. (Specification, p. 10, l. 24; p. 12, ll. 10-15; figure 4, step 40). The computer program product further includes third instructions for switching to a standby adapter handled by the device driver responsive to detecting the failure, wherein the standby adapter uses the data in the data queue. (Specification, p. 10, l. 24; p. 12, ll. 10-15; figure 4, step 404).

D. CLAIM 22 - INDEPENDENT

The subject matter of claim 22 is directed to a server data processing for obtaining cultural context information from a client (Specification, p.5, ll. 4-16; figure 1, all elements). The server data processing system includes a bus system. (Specification, p.5, l. 4-p. 6, l. 2; figure 1, elements 106, 110-128). The server data processing system further includes a communications unit connected to the bus system. (Specification, p. 5, ll. 17-24; figure 1, elements 118, 120). The server data processing system further includes a memory connected to the bus system, wherein the memory includes a set of instructions. (Specification, p. 5, ll. 11-13; figure 1, elements 108, 109;). The server data processing system further includes a processing unit connected to the bus system. (Specification, p. 5, ll. 7-11; figure 1, elements 102, 104). The processing unit executes instructions for a device driver to queue data in a data queue used by the primary adapter. (Specification, p. 9 ll. 20-25; p.11, ll. 19-20; fig. 3, step 316). The processing unit further monitors the primary adapter for the failure. (Specification, p. 10, l. 24; p. 12, ll. 10-15; figure 4, step 40). The processing unit further switches to a standby adapter handled by the device driver in response to detecting the failure, wherein the standby adapter uses the data in the data queue. (Specification, p. 10, l. 24; p. 12, ll. 10-15; figure 4, step 404).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The grounds of rejection to review on appeal are as follows:

A. GROUND OF REJECTION 1

Whether the examiner failed to state a *prima facie* obviousness rejection against claims 1-22 over *Mahalingham et al.*, Means for Allowing Two or More Network interface Controller Cards to Appear as One Card to an Operating System, U.S. Patent 6,314,525 (November 6, 2001) (hereinafter "*Mahalingham*") in view of *Kurapati et al.*, Data Replication for Redundant Network Components, U.S. Patent 7,007,190 (February 28, 2006)(hereinafter "*Kurapati.*") under 35 U.S.C. § 103(a).

ARGUMENT

A. GROUND OF REJECTION 1 (Claims 1-22)

The Examiner rejects claims 1-22 as obvious over *Mahalingham* in view of *Kurapati.* Appellants request that the Board of Patent Appeals and Interferences overturn this rejection and direct the examiner to allow the claims.

A.1. Claims 1-7

A.1.i. Proposed Combination Does Not Teach all of the Features of Claim 1

The examiner rejects claims 1-7 as anticipated by *Mahalingham* in view of *Kurapati.* Claim 1 is a representative claim in this grouping of claims. Claim 1 is as follows:

> 1. A method in a device driver for handling a failure of a primary adapter in a data processing system, the method comprising:
>   queuing data in a data queue used by the primary adapter;
>   monitoring the primary adapter for the failure; and
>   responsive to detecting the failure, switching to a standby adapter handled by the device driver, wherein the standby adapter uses the data in the data queue.

If the Patent Office does not produce a *prima facie* case of unpatentability, then without more the applicant is entitled to grant of a patent. *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Grabiak*, 769 F.2d 729, 733, 226 U.S.P.Q. 870, 873 (Fed. Cir. 1985). A *prima facie* case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994).

The examiner has failed to state a *prima facie* obviousness rejection against claim 1 because the proposed combination of *Mahalingham* in view of *Kurapati* does not teach all of the features of claim 1. As admitted by the examiner, *Mahalingham* does not teach or suggest all of the features of claim 1. Furthermore, as shown below, *Kurapati* does not cure the deficiencies of *Mahalingham* in this regard. For this reason alone, the proposed combination does not teach all of the features of claim 1. Accordingly, the proposed combination does not teach all of the claimed features and the examiner failed to state a *prima facie* obviousness rejection against claim 1.

With regard to the data queuing step of claim 1, the examiner states:

> Mahalingham et al. does not explicitly teach the method of queuing data in a data queue used by the primary adapter and in response to detecting a failure wherein the standby adapter uses the data in the data queue.
>
> Kurapati et al. disclosed the method of data replication for redundant network components by implementing a shared memory queue (fig. 6, 120, col. 9 lines 10-19) and heap memory queue of message queue (fig. 6, 110, col. 9 lines 10-12), which may be designated to provide data to a specific process (fig. 2, 50). The shared memory also provides efficient communication between processes by allowing one process to write data to shared memory and another process to read the data from shared memory (fig. 2, 50, 51, col. 11 lines 55-58). Furthermore, in the case of a detection of an internal fault or defective process (col. 13 lines 49-54) the standby component uses the data of the active component by receiving a replication of that data before it enters active mode. The data replicator sends replication data to network component 12b (fig. 11, 230, col. 13 lines 56-63).
>
> It would have been obvious to one of ordinary skill in the art to modify the method of Mahalingham et al. by adding Kurapati et al. method of data replication for redundant network components. A person of ordinary skill in the art at the time of appellants' invention would have been motivated to make the modification because it would provide a reliable data replication (col. 2 lines 18-20).

Final Office Action of September 13, 2006, pp. 3-4.

The examiner admits, and Appellants agree, that *Mahalingham* does not teach the claimed feature of, "queuing data in a data queue *used by the primary adapter*," as in claim 1. Additionally, given the lack of disclosure in *Mahalingham* in this regard, *Mahalingham* also does not suggest this claimed feature.

Nevertheless, the examiner asserts that *Kurapati* teaches this claimed feature. However, the examiner's assertions regarding *Kurapati vis-à-vis* claim 1 are erroneous. *Kurapati* teaches a system for routing voice calls. As part of that system, redundant software components exchange replicated information so that calls are not dropped in case one component fails or is dropped. However, the software components are not adapters as claimed. Thus, *Kurapati* does not teach, "queuing data in a data queue *used by the primary adapter*," as claimed. Similarly, *Kurapati* does not teach, "responsive to detecting the failure, switching to a standby adapter handled by the device driver, *wherein the standby adapter uses the data in the data queue*," as claimed.

Appellants now address the examiner's citations to *Kurapati* to show that the examiner's assertions are erroneous. The examiner first cites the following:

> FIG. 6 is a block diagram illustrating examples of shared memory queue 120 and heap memory queue 122 of message queue 110 of FIG. 5. Message queue 110 may be designated to provide data to a specific process 50. Message queue 110 receives a sequence of pointers to message buffers 111 and 112 that store data. The pointers to message buffers 111 are stored in heap memory queue 122, and the pointers to message buffers 112 are stored in shared memory queue 120. Shared memory queue 120 and heap memory queue 122 include fields that record the sequence of pointers.

Kurapati, col. 9, ll. 10-19.

Kurapati teaches that a message queue is designated to provide data to specific processes 50. Pointers to a first set of message buffers are received in the message queue and are stored in a heap memory queue, whereas the pointers to a second set of memory buffers are stored in shared memory queue. However, *Kurapati* discloses that processes 50 are active software processes, not adapters as claimed. For example, *Kurapati* states:

> Network component 12 includes one or more processes 50, a shared memory 51, platform 52, and an operating system 54. *Processes 50 may comprise any number of software applications that perform the operations of network component 12.* A process 50 may access a designated process library 56 or a process library 56 for another process 50. A process library 56 stores software code that may be used to perform the operations. A process 50 may use one or more process threads 53 to provide concurrent processing.

Kurapati, col. 5, l. 64 through col. 6, l. 5 (emphasis added).

In contrast, an adapter is a hardware component having a specific function, as described in Appellants' specification. Thus, data queues in *Kurapati* are not used in any kind of adapter, much less a primary adapter, as claimed. Similarly, no standby adapter in *Kurapati* uses the data in the queue, as claimed.

The examiner also cites the following portion of *Kurapati*:

> Shared memory 51 provides for efficient communication between processes 50 by allowing one process 50 to write data to shared memory 51, and another process to read the data from shared memory 51.

Kurapati, col. 11, ll. 55-58.

However, again, *Kurapati* discloses that shared memory allows for communication between *processes*, not adapters, as claimed. The examiner also refers to the following portion of *Kurapati*:

> Network component 12a may switch from an active mode to a standby mode, and network component 12b may switch from a standby mode to an active mode if, for example, network component 12a detects an internal fault or a defective process 50. …
>
> The method begins at step 230, where network component 12a is operating in an active mode, and mate network component 12b is operating in a standby mode. Data replicator 62a sends replication data to network component 12b. Data replicator 62a stops sending replication data at step 232, and network component 12a enters a transient standby mode.

Kurapati, col. 13, ll. 46-63.

Kurapati teaches that a network component may switch from active mode to standby mode. Data replicators send data to the other network component. However, *Kurapati* does not disclose an adapter. Instead, as cited above, *Kurapati* describes a network component as software components. Software components are not adapters, as claimed. Thus, *Kurapati* does not teach, "queuing data in a data queue *used by the primary adapter*," as claimed. Similarly, *Kurapati* does not teach, "responsive to detecting the failure, switching to a standby adapter handled by the device driver, *wherein the standby adapter uses the data in the data queue*," as claimed. Because *Kurapati* is devoid of disclosure in regard to these features, nothing in *Kurapati* suggests these claimed features.

Thus, *Kurapati* does not teach or suggest all of the features of claim 1. As established above, *Mahalingham* also does not teach or suggest the same features of claim 1. For this reason, the proposed combination, considered as a whole, does not teach or suggest all of the features of claim 1. Therefore, under the standards of *In re Lowry*, the examiner failed to state a *prima facie* obviousness rejection against claim 1 or any of the other claims in this grouping of claims.

A.1.ii. The Examiner Failed To State a Proper Teaching, Suggestion, or Motivation To Combine the References

Additionally, the examiner failed to state a proper teaching, suggestion, or motivation to combine the references. Regarding a reason to combine the references, the examiner states that:

> It would have been obvious to modify the method of Mahalingham et al, by adding Kurapati et al. method of data replication for redundant network components. A person of ordinary skill in the art at the time of applicant's invention would have been motivated to make the modification because it would provide a reliable data replication (col. 2 lines 18-20).

Final Office Action of September 13, 2006, p. 4.

The examiner states that the motivation to modify *Mahalingham* would be to provide reliable data replication. However, the proposed advantage does not exist, *vis-à-vis* claim 1, given the vast differences between *Mahalingham, Kurapati,* and claim 1.

Mahalingham, as shown above, is related to software and not to the claimed primary adapters, which are hardware. Specifically, *Mahalingham* discloses a method and apparatus for allowing a plurality of network interface controllers (NICs) to appear as a single NIC to an operating system. A first interface bound to at least one transport protocol stack and a plurality of network adapters is coupled to a computer network that supports recovery from network adapter and connection failure. A second interface is bound to the plurality of network adapters, as well as a mechanism coupled to the first interface. The second interface receives a first MAC-level packet from a transport protocol stack through the first interface and forwards the first MAC-level packet through the second interface to a network adapter in a protocol independent matter. A mechanism coupled to the first interface and the second interface receives the second packet from a network adapter through the second interface and forwards the second packet through the first interface to a transport protocol stack.

In contrast, *Kurapati* teaches data replication among software components. *Mahalingham* teaches a method of allowing network interface controllers to appear as a single network interface controller. Given the stark differences between the references, *Mahalingham* has no need for data replication among software components. The fact that *Kurapati* teaches an advantage regarding data replication has *nothing* to do with the methods and devices taught in *Mahalingham*. Thus, the examiner's asserted motivation to combine the references does not exist. Given the differences between claim 1 and both *Mahalingham* and *Kurapati*, utterly no motivation exists in fact to combine the references to achieve the invention of claim 1.

Instead, the examiner appears to have created the asserted motivation based on the examiner's personal opinion, which constitutes impermissible hindsight. For this reason, the examiner failed to state a proper teaching, suggestion, or motivation to combine the references. Accordingly, the examiner failed to state a *prima facie* obviousness rejection against claim 1 or against the remaining claims in this grouping of claims.

A.1.iii. No Teaching, Suggestion, or Motivation Exists to Combine the References Because Each Reference Represents a Complete Solution to the Problem That Each Solves

The examiner has failed to state a prima facie obviousness rejection against claim 1 because no proper teaching, suggestion, or motivation exists to combine the references. No proper teaching or motivation exists to combine the references because both *Mahalingham* and *Kurapati* represent complete solutions to the problems each solves.

Mahalingham discloses a method and apparatus for allowing a plurality of network interface controllers (NICs) to appear as a single NIC to an operating system. A first interface bound to at least one transport protocol stack and a plurality of network adapters is coupled to a computer network that supports recovery from network adapter and connection failure. A second interface is bound to the plurality of network adapters, as well as a mechanism coupled to the first interface. The second interface receives a first MAC-level packet from a transport protocol stack through the first interface and forwards the first MAC-level packet through the second interface to a network adapter in a protocol independent matter. A mechanism coupled to the first interface and the second interface receives the second packet from a network adapter through the second interface and forwards the second packet through the first interface to a transport protocol stack. Specifically, *Mahalingham* provides that:

> The method comprises: (1) sending a packet from the server through the primary adapter across the network to a secondary adapter; (2) attempting to receive the packet at the secondary adapter; (3) sending a packet from the server through a secondary adapter to the primary adapter; (4) attempting to receive the packet from the primary adapter; (5) if the packet sent by the primary adapter is not received by the secondary adapter, and if packet by the secondary adapter is not received by the

> primary adapter, the system concludes that the primary adapter has failed and converts a secondary adapter into a replacement primary adapter.

Mahalingham, col. 2, l. 59-col. 3, l. 3.

On the other hand, *Kurapati* teaches a system for routing voice calls. Redundant software components exchange replicated information so that calls are not dropped in case one component fails or is dropped. A message queue is designated to provide data to specific processes. Pointers to a first set of message buffers are received in the message queue and are stored in a heap memory queue, whereas the pointers to a second set of memory buffers are stored in shared memory queue. *Kurapati* teaches that a network component may switch from active mode to standby mode. Data replicators send data to the other network component. *Kurapati* provides that:

> An entry of a replication table of a shared memory database is assigned to a stable call. The replication table is associated with a network component operable to process the stable call. Data associated with the stable call is received and is stored in the shared memory database. The entry is updated in response to storing the data. The data is sent to a mate network component of the network component, and the entry is updated in response to sending the data.

Kurapati, col. 1, l. 62-col. 2, l. 3.

The entire principle of operation of *Mahalingham* provides that packet information is rerouted to a secondary adapter upon failure of receipt by a primary adapter. By converting a secondary adapter to a replacement primary adapter, *Mahalingham* provides fault tolerance and load sharing across multiple NICs. *Mahalingham* has no need for redundant software components to exchange replicated information because *Mahalingham* reroutes the same information packet from one adapter to another. Providing a replication table of a shared memory database as described in *Kurapati* would mean modifying, altering or replacing the principle of operation of *Mahalingham's* system.

Each reference provides a complete solution to the problem that each reference represents, one of ordinary skill would have no reason to combine or otherwise modify the references in the proposed manner. Additionally, combining *Mahalingham* with *Kurapati* would change the principle of operation of *Mahalingham*. For these reasons, no teaching, suggestion, or motivation exists to combine the references to achieve the invention of claim 1. Accordingly, the examiner has failed to state a *prima facie* obviousness rejection against claim 1 or against other claims in this grouping of claims.

A.2. Claims 8-14

A.2.i. Proposed Combination Does Not Teach all of the Features of Claim 8

The examiner rejects claims 8-14 as anticipated by *Mahalingham* in view of *Kurapati.* Claim 8 is a representative claim in this grouping of claims. Claim 8 is as follows:

> 8. A data processing system for handling a failure of a primary adapter in a data processing system, the data processing system comprising:
>
> queuing means for queuing data in a data queue used by the primary adapter;
>
> monitoring means for monitoring the primary adapter for the failure; and
>
> switching means for switching to a standby adapter handled by the device driver responsive to detecting the failure, wherein the standby adapter uses the data in the data queue.

If the Patent Office does not produce a *prima facie* case of unpatentability, then without more the applicant is entitled to grant of a patent. *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Grabiak*, 769 F.2d 729, 733, 226 U.S.P.Q. 870, 873 (Fed. Cir. 1985). A *prima facie* case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994).

The examiner has failed to state a *prima facie* obviousness rejection against claim 8 because *Mahalingham* in view of *Kurapati,* considered as a whole, does not teach all of the features of claim 8. *Mahalingham* does not teach or suggest all of the features of claim 8. Furthermore, *Kurapati* does not cure the deficiencies of *Mahalingham* in this regard. The examiner admits that *Mahalingham et al.* does not explicitly teach the method of queuing data in a data queue used by the primary adapter and in response to detecting a failure wherein the standby adapter uses the data in the data queue. As established *supra*, *Kurapati* does not teach, "queuing data in a data queue *used by the primary adapter*," as claimed. Similarly, *Kurapati* does not teach, "responsive to detecting the failure, switching to a standby adapter handled by the device driver, *wherein the standby adapter uses the data in the data queue*," as claimed. For this reason alone, the proposed combination does not teach all of the features of claim 8. Accordingly, the proposed combination does not result in the claimed invention and the examiner has failed to state a *prima facie* obviousness rejection against claim 8.

A.2.ii. The Examiner Failed To State a Proper Teaching, Suggestion, or Motivation To Combine the References

As shown above with respect to claim 1, the examiner failed to state a *prima facie* obviousness rejection because the examiner's proposed motivation does not exist. The same fact applies to the rejection of claim 8. Therefore, the examiner has also failed to state a *prima facie* obviousness rejection against claim 8 and against the remaining claims in this grouping of claims.

A.2.iii. No Teaching, Suggestion, or Motivation Exists to Combine the References Because Each Reference Represents a Complete Solution to the Problem That Each Solves

Additionally, the examiner has failed to state a *prima facie* obviousness rejection against claim 8 because no proper teaching, suggestion, or motivation exists to combine the references. No proper teaching or motivation exists to combine the references because both *Mahalingham* and *Kurapati* represent complete solutions to the problems each solves.

Because each reference provides a complete solution to the problem that each reference represents, one of ordinary skill would have no reason to combine or otherwise modify the references. For this reason, no teaching, suggestion, or motivation exists to combine the references to achieve the invention of claim 8. Accordingly, the examiner has failed to state a *prima facie* obviousness rejection against claim 8.

A.3. Claims 15-21

A.3.i. Proposed Combination Does Not Teach all of the Features of Claim 15

The examiner rejects claims 15-21 as anticipated by *Mahalingham* in view of *Kurapati.* Claim 15 is a representative claim in this grouping of claims. Claim 15 is as follows:

> 15. A computer program product in a recordable-type medium for handling a failure of a primary adapter in a data processing system, the computer program product comprising:
>
> first instructions for queuing data in a data queue used by the primary adapter;
>
> second instructions for monitoring the primary adapter for the failure; and
>
> third instructions for switching to a standby adapter handled by the device driver responsive to detecting the failure, wherein the standby adapter uses the data in the data queue.

If the Patent Office does not produce a *prima facie* case of unpatentability, then without more the applicant is entitled to grant of a patent. *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Grabiak*, 769 F.2d 729, 733, 226 U.S.P.Q. 870, 873 (Fed. Cir. 1985). A *prima facie* case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994).

The examiner has failed to state a *prima facie* obviousness rejection against claim 15 because *Mahalingham* in view of *Kurapati* does not teach all of the features of claim 15. *Mahalingham* does not teach or suggest all of the features of claim 15 at least for the reasons presented above. Furthermore, *Kurapati* does not cure the deficiencies of *Mahalingham* in this regard. The examiner admits that *Mahalingham et al.* does not explicitly teach the method of queuing data in a data queue used by the primary adapter and in response to detecting a failure wherein the standby adapter uses the data in the data queue. As established supra, *Kurapati* does not teach, "queuing data in a data queue *used by the primary adapter*," as claimed. Similarly, *Kurapati* does not teach, "responsive to detecting the failure, switching to a standby adapter handled by the device driver, *wherein the standby adapter uses the data in the data queue*," as claimed. For this reason alone, the proposed combination does not teach all of the features of claim 15. Accordingly, the proposed combination does not result in the claimed invention and the examiner has failed to state a *prima facie* obviousness rejection against claim 15.

A.3.ii. The Examiner Failed To State a Proper Teaching, Suggestion, or Motivation To Combine the References

As shown above with respect to claim 1, the examiner failed to state a *prima facie* obviousness rejection because the examiner's proposed motivation does not exist. The same fact applies to the rejection of claim 15. Therefore, the examiner has also failed to state a *prima facie* obviousness rejection against claim 15 and against the remaining claims in this grouping of claims.

A.3.iii. No Teaching, Suggestion, or Motivation Exists to Combine the References Because Each Reference Represents a Complete Solution to the Problem That Each Solves

The examiner has failed to state a *prima facie* obviousness rejection against claim 15 because no proper teaching, suggestion, or motivation exists to combine the references. No proper teaching or motivation exists to combine the references because both *Mahalingham* and *Kurapati* represent complete solutions to the problems each solves.

Because each reference provides a complete solution to the problem that each reference represents, one of ordinary skill would have no reason to combine or otherwise modify the references. For this reason, no teaching, suggestion, or motivation exists to combine the references to achieve the invention of claim 15. Accordingly, the examiner has failed to state a *prima facie* obviousness rejection against claim 15.

A.4. Claim 22

A.4.i. Proposed Combination Does Not Teach all of the Features of Claim 22

The examiner rejects claim 22 as anticipated by *Mahalingham* in view of *Kurapati.* Claim 22 is a representative claim in this grouping of claims. Claim 22 is as follows:

> 22. A server data processing for obtaining cultural context information from a client, the server data processing system comprising:
> a bus system;
> a communications unit connected to the bus system;
> a memory connected to the bus system, wherein the memory includes a set of instructions; and

> a processing unit connected to the bus system, wherein the processing unit executes instructions for a device driver to queue data in a data queue used by a primary adapter, monitor the primary adapter for the failure and, switch to a standby adapter handled by the device driver in response to detecting the failure, wherein the standby adapter uses the data in the data queue.

If the Patent Office does not produce a *prima facie* case of unpatentability, then without more the applicant is entitled to grant of a patent. *In re Oetiker*, 977 F.2d 1443, 1445, 24 U.S.P.Q.2d 1443, 1444 (Fed. Cir. 1992); *In re Grabiak*, 769 F.2d 729, 733, 226 U.S.P.Q. 870, 873 (Fed. Cir. 1985). A *prima facie* case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994).

The examiner has failed to state a *prima facie* obviousness rejection against claim 22 because *Mahalingham* in view of *Kurapati* does not teach all of the features of claim 22. *Mahalingham* does not teach or suggest all of the features of claim 22. Furthermore, *Kurapati* does not cure the deficiencies of *Mahalingham* in this regard. The examiner admits that *Mahalingham et al.* does not explicitly teach the method of queuing data in a data queue used by the primary adapter and in response to detecting a failure wherein the standby adapter uses the data in the data queue. As established *supra*, *Kurapati* does not teach, "queuing data in a data queue *used by the primary adapter*," as claimed. Similarly, *Kurapati* does not teach, "responsive to detecting the failure, switching to a standby adapter handled by the device driver, *wherein the standby adapter uses the data in the data queue*," as claimed. For this reason alone, the proposed combination does not teach all of the features of claim 22. Accordingly, the proposed combination does not result in the claimed invention and the examiner has failed to state a *prima facie* obviousness rejection against claim 22.

A.4.ii. The Examiner Failed To State a Proper Teaching, Suggestion, or Motivation To Combine the References

As shown above with respect to claim 1, the examiner failed to state a *prima facie* obviousness rejection because the examiner's proposed motivation does not exist. The same fact applies to the rejection of claim 22. Therefore, the examiner has also failed to state a *prima facie* obviousness rejection against claim 22.

A.4.iii. No Teaching, Suggestion, or Motivation Exists to Combine the References Because Each Reference Represents a Complete Solution to the Problem That Each Solves

The examiner has failed to state a *prima facie* obviousness rejection against claim 22 because no proper teaching, suggestion, or motivation exists to combine the references. No proper teaching or motivation exists to combine the references because both *Mahalingham* and *Kurapati* represent complete solutions to the problems each solves.

Because each reference provides a complete solution to the problem that each reference represents, one of ordinary skill would have no reason to combine or otherwise modify the references. For this reason, no teaching, suggestion, or motivation exists to combine the references to achieve the invention of claim 22. Accordingly, the examiner has failed to state a *prima facie* obviousness rejection against claim 22.

B. CONCLUSION

As shown above, the examiner has failed to state a *prima facie* obviousness rejection against any of the claims. Therefore, Appellants request that the Board of Patent Appeals and Interferences reverse the rejections. Additionally, Appellants request that the Board direct the examiner to allow the claims

/Theodore D. Fay III/
Theodore D. Fay III
Reg. No. 48,504
YEE & ASSOCIATES, P.C.
PO Box 802333
Dallas, TX 75380
(972) 385-8777

TF/BW

CLAIMS APPENDIX

The texts of the claims involved in the appeal are:

1. A method in a device driver for handling a failure of a primary adapter in a data processing system, the method comprising:

queuing data in a data queue used by the primary adapter;

monitoring the primary adapter for the failure; and

responsive to detecting the failure, switching to a standby adapter handled by the device driver, wherein the standby adapter uses the data in the data queue.

2. The method of claim 1, wherein the failure is an occurrence of at least one of a network problem and a port problem.

3. The method of claim 1, wherein the primary adapter is on a first port and the standby adapter is on a second port and wherein the switching step comprises:

switching from the first port to the second port to switch to the standby adapter.

4. The method of claim 3, wherein the first port is assigned an alternative media access control address prior to a switch from the primary adapter to the standby adapter and wherein the switch from the first port to the second port is made by assigning the second port to an alternative media access control address.

5. The method of claim 3 further comprising:

initiating a soft reset of the first port.

6. The method of claim 1, wherein the primary adapter is a network adapter.

7. The method of claim 1, wherein the primary adapter is a graphics adapter.

8. A data processing system for handling a failure of a primary adapter in a data processing system, the data processing system comprising:

queuing means for queuing data in a data queue used by the primary adapter;

monitoring means for monitoring the primary adapter for the failure; and

switching means for switching to a standby adapter handled by the device driver responsive to detecting the failure, wherein the standby adapter uses the data in the data queue.

9. The data processing system of claim 8, wherein the failure is an occurrence of at least one of a network problem and a port problem.

10. The data processing system of claim 8, wherein the primary adapter is on a first port and the standby adapter is on a second port and wherein the switching means comprises:

means for switching from the first port to the second port to switch to the standby adapter.

11. The data processing system of claim 10, wherein the first port is assigned an alternative media access control address prior to a switch from the primary adapter to the standby adapter and wherein the switch from the first port to the second port is made by assigning the second port to an alternative media access control address.

12. The data processing system of claim 10 further comprising:

initiating means for initiating a soft reset of the first port.

13. The data processing system of claim 8, wherein the primary adapter is a network adapter.

14. The data processing system of claim 8, wherein the primary adapter is a graphics adapter.

15. A computer program product in a recordable-type medium for handling a failure of a primary adapter in a data processing system, the computer program product comprising:

first instructions for queuing data in a data queue used by the primary adapter;

second instructions for monitoring the primary adapter for the failure; and

third instructions for switching to a standby adapter handled by the device driver responsive to detecting the failure, wherein the standby adapter uses the data in the data queue.

16. The computer program product of claim 15, wherein the failure is an occurrence of at least one of a network problem and a port problem.

17. The computer program product of claim 15, wherein the primary adapter is on a first port and the standby adapter is on a second port and wherein the second instructions comprise:

Sub-instructions for switching from the first port to the second port to switch to the standby adapter.

18. The computer program product of claim 17, wherein the first port is assigned an alternative media access control address prior to a switch from the primary adapter to the standby adapter and wherein the switch from the first port to the second port is made by assigning the second port to an alternative media access control address.

19. The computer program product of claim 17 further comprising:

fourth instructions for initiating a soft reset of the first port.

20. The computer program product of claim 15, wherein the primary adapter is a network adapter.

21. The computer program product of claim 15, wherein the primary adapter is a graphics adapter.

22. A server data processing for obtaining cultural context information from a client, the server data processing system comprising:

a bus system;

a communications unit connected to the bus system;

a memory connected to the bus system, wherein the memory includes a set of instructions; and

a processing unit connected to the bus system, wherein the processing unit executes instructions for a device driver to queue data in a data queue used by a primary adapter, monitor the primary adapter for the failure and, switch to a standby adapter handled by the device driver in response to detecting the failure, wherein the standby adapter uses the data in the data queue.

EVIDENCE APPENDIX

There is no evidence to be presented.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings.